



GREETINGS,  
USER...

## ADDRESS

CHRIS HESTER,  
3 WEST LANE,  
BAILDON,  
NEAR SHIPLEY,  
WEST YORKSHIRE,  
BD17 5HD, UK.

## SUBSCRIPTIONS

You can order one issue or as many as you like.

- £1.25 Britain (2nd class)
- £1.31 Britain (1st class)
- £1.42 Elsewhere (Seamail)
- £1.47 Europe (Airmail)
- £1.74 Zone 1 (Airmail)
- £1.83 Zone 2 (Airmail)

Zone 1 = Canada, Egypt, Falklands, Saudi Arabia, Seychelles, USA...

Zone 2 = Australia, New Zealand...

## PAYMENT

Please pay by cheque, postal order or International Giro, or you can send British decimal stamps for small amounts. All cheques should be made payable to C. HESTER please, NOT the magazine's name.

## BACK ISSUES

These cost the same as the current issue.

## ADVERTISING

These are the costs:

- £5.00 inside page
- £6.00 back cover
- £2.50 half a page
- £1.25 quarter page

## LETTERS

I'll be only too pleased to read your letters, but if you want a personal reply, please enclose a stamped addressed envelope. If you don't want your letter printed in this magazine you must write 'NOT FOR PUBLICATION' on the top.

## CONTRIBUTIONS

These are always welcome, and keep the magazine going! I'd prefer your articles typed with a small margin (but hand-written ones are also welcome!). Make sure the ink is as dark as possible, so the text will photocopy clearly. You may like to send me a drawing or a cartoon too.

## FORMAT

This is an amateur magazine printed on an irregular basis as a non-profit hobby.

## COPYRIGHT

This magazine is (C) Christopher Hester (Editor) 1993, except for items credited to another author who retains full copyright on their material, and are free to use it elsewhere.

## PRODUCTION

I, the Editor, design the whole magazine and paste up the artwork.

PUBLISHED BY  
LYNSOFT.

## FINALLY...

Thank you for seeking out this magazine and for taking the time to read it.

ISSUE 20 ~~WTO 8/93~~

V1.1 7/93

## C O N T E N T S - C O N T E N T E D ?

- Page 4 - EDIT TORY ALL - The Editor on the magazine - but which one?
- Page 5 - STRING 'EM UP - Robin Rawson-Tetley on BASIC strings.
- Page 6 - THE TAXMAN CHRONICLES - Part 2 - Steve Clay continues writing his game.
- Page 7 - PUZZLED - And Steve Clay tells how one of the puzzles came into being.
- Page 8 - WHOLE LOTTA STAC - George March offers help to one reader using STAC.
- Page 10 - ADVENTURE LINK 2 - Reviewed here for us by George, for all Speccies.
- Page 13 - AWARDS - Big Ed lists his faves of last year. What were yours?
- Page 14 - FURTHER INTO P.D. - George March continues his look at Public Domain.
- Page 18 - HOW TO MAKE A THINKY (PART IV) - Stephen Groves' adventure writer grows.
- Page 26 - STAC ANIMATION - yes it is possible! Dave Blower reveals the secrets.
- Page 29 - AMOS AVENUE - John Ferris tackles the plot of an adventure.
- Page 32 - PAW TECHNICAL GUIDE - Steve Clay carries on where Gilsoft left off.
- Page 33 - FOOD FOR THOUGHT - Samantha Fox reviews Madonna's book. And more.
- This issue's cover was drawn by the Editor, who'd like to thank himself for that.
- APOLOGY!** In the last issue of 'Adventure Workshop' I forgot to credit the article 'Writing Tips' to Steve Clay. My apologies go to Steve, his family, relatives, everyone who knows him, and his pet goldfish. Sorry!

## EDIT TORY ALL

Welcome to a new format Adventure Coder! Or should I say welcome back to the old format! I've found that there will not be enough material to keep my other magazine Adventure Workshop going. So rather than do away with the articles I do have for 16-bit computers, I've decided to return these to the pages of this magazine, which used to cater for 16-bit machines before anyway! So now Adventure Coder is a multi-format (16-bit and 8-bit) magazine again! Yet far from destroying the recent flavour of Coder, this can, I feel, only add to it. Many users upgrade to an ST, PC or Amiga, but also retain their much-loved Spectrum, Amstrad or C64! Whilst others are keen to convert from, say, Spectrum to ST. So clearly Coder could cater for a larger audience now than ever before. What's more, consider this: I was having to alternate publication between Coder and Workshop, giving a minimum release date of two months between each issue - one month for a Coder issue, then a month while I released Workshop. But now, just as before, I can concentrate fully on just the one magazine, meaning it can effectively be put out twice as soon! And it'll be a lot less confusing for me, trying to run two magazines at once! (I'll no longer need to wonder if a new article is right for one magazine only, and which one, or if it might go in both, meaning it would have to be reprinted or copied.)

So in this issue you'll find material for STAC users on the ST, a review of a tapezine (?) for the Spectrum, an article for AMOS users on the Amiga, more from Steve Clay on his efforts to program his adventure, two general articles on adventures, an advanced article on getting animation into STAC adventures (:), more about the P.D. scene, the latest part of Stephen Groves' look into machine code, a technical guide index for PAW, and more! Not bad considering the adventure scene is supposed to be 'dead'!

The only thing you'll find is that I've had to postpone the Letters Page until next issue to get everything in. Sorry about that, but I can guarantee a bumper set of letters next time!

Adventures aside, I took my first driving test last November! What a nerve-racking (or is it nerve-wrecking?) experience! So much so that I admit I failed! But it doesn't matter anymore, because I took another test in January, and this time passed! At first I didn't believe it, because I was sure I'd messed up the test in the first five minutes! I think that made me relax more, as I thought it was all over after that. How wrong I was when I was told I'd passed! Hurrah! So it means no more expensive lessons, and I can drive on my own now! By chance my Mum's old car bit the dust so for weeks she was without one I could drive! Now she has a brand new Polo Genesis! (And yea, it is named after the group! Volkswagen actually sponsored the last Genesis tour!) Not that she got this deliberately - she always gets a Polo! I only hope I don't wreck it!

Next issue I've got articles for users of BASIC, STAC, GAC, and more! And if your system isn't covered, write in and let me know, and I'll see what I can do! I want to see also material coming in for the Sam Coupé, PC, and other machines. And I want articles about using Talespin, STOS, even the 3D Construction Kit 1 and 2. Plus I need more articles on grammar and spelling in adventures, ideas for plots, objects and puzzles, characters, the lot. I know some of you have some excellent ideas, so let's hear about them. Or simply jot down a line or two about adventures in general - maybe you're writing the next masterpiece and want to tempt us with a few of the details? Go on!

Don't forget there's the chance to write a whole page yourself, no matter what the subject is, to air your views on something - maybe the pit closure programme? Or the privatisation of British Rail? Or the price of CDs? So long as it's nothing libellous, I hope to print whatever you send me. Mark your article 'Your Say'.

I trust you will enjoy this issue then - the first of 1993! Yours, the Editor.

## String 'em Up

by Robin Rawson-Tetley

With my adventure system featuring in these very pages, I felt that it could have been easily converted to other machines, such as the Commodore 64 and IBM PC (QBASIC - want any PC articles, Chris?), but only if the person converting had a knowledge of Spectrum BASIC, which for some reason is totally unlike other BASICs in it's string handling methods.

On the Spectrum, if you wish to check for a particular character in a string, say, you have to do this..

Imagine your string is in a\$, and you want to test if character 5 is the letter "Y", you would have to enter "IF a\$(5)="Y\"", simple, eh?

If you wanted to do this on a 64, you would have to enter "IF MID\$(a\$,5,1)="Y"" - This MID\$ command is in the format "(string\$,starting character,number of characters)", so you can see that it takes one character from a\$, starting at character 5.

The Spectrum does have it's own MID\$ function - imagine you want 3 characters from z\$, starting at character 4.

On a Spectrum, you would enter "a\$=z\$(4 TO )( TO 3)" or you could use "a\$=z\$(4 TO 7)". On a 64, you would use "a\$=MID\$(z\$,4,3)".

The other universal-not-on-the-Spectrum string handling commands are LEFT\$ and RIGHTS\$.

On a 64, if you had "LEFT\$ (a\$,3)", on a Spectrum, you would have to use "a\$( TO 3)" and if you had, on a 64, "RIGHT\$ (a\$,4)", you would have to use "a\$((LEN a\$)-4) TO )" (see how you subtract the number of characters from the end of the string and take the string from there to the end).

There is one other major command on the Spectrum in adventures, which is not called the same on any other machine, that command is CODE, it is used to find the ASCII value of a character, on the other machines, it simply has a different name - ASC, so "CODE ("B")" on a Spectrum is "ASC ("B")" on a 64.

Hope this is of use.

## THE TAXMAN CHRONICLES - Part 2

by Steve Clay

The first entry in this trio of entries ended like a lesson in using PAW. I'll try and curb this tendency in future.

As I write this, Tax Returns is undergoing its first playtest. So I sit waiting the arrival of the report with the numerous bugs I have no doubt missed.

Last time I spent a lot of the time explaining the Light rooms puzzle and the routines behind it. The light rooms were intended for just inside the mine, however a change of heart now leaves the light rooms as a separate building. Also I was expecting to use a lift to connect the various levels of the mine. More realistically I felt would be the inclusion of a mine truck. The truck is voice operated and using a simple numerical command the player can get to one of the five sections of the mine. Five? With seven dwarves and Snow White you would expect eight sections, but some of the characters live in the same section.

Anyone who has played The Taxman Cometh, and by all accounts it isn't many, will know that I write puzzle based games. I use little in the way of purple prose in my location descriptions. That is not to say that I skimp on my locations, it is more to do with the fact that I find I can't take anything too seriously. I'd much rather have a giggle.

I started writing a 'serious' game once. Very quickly, however, I found myself writing things like;  
[East/west passage]

The passage leads east and west. Well it would wouldn't it. I mean, if it went north and south, it wouldn't be described as an east/west passage would it?

This is how many of my locations end up. I include all the necessary details within the text, but try and keep it all tongue-in-cheek. A point of interest, during Tax Returns one of the longest location descriptions is for a narrow passage that the player lands in. It serves no other purpose.

Anyway to proceed. Once the mine truck was rattling around like a good 'un, I could start fitting the residents into their relevant sections. I had several good puzzles from earlier abortive attempts. These were used to the individual characters they suited. Obviously there were changes in objects and settings, but the actual puzzles had already been tested and found acceptable, by me anyway. Some of you may be thinking; "If the puzzles are so good, how come the other games were scrapped?" Simple, it was the actual design of the locations and the way the puzzles hung together that was at fault not the actual puzzle design. (I hope!)

The final eight debtors are; Snow White (Not copyright of Walt Disney!), Gadget, Trapper, Blotto, Potboy, Parser, Banker and Nomarks. I have made some of these simple to solve, to keep the player interested and the others are progressively harder.

The first puzzle written for a character was Blotto's barrel puzzle. This involved a series of moves involving a barrel outside Blotto's door. The player has a move limit after knocking on the door. I dislike timed puzzles when the time is a forceful presence or the wrong move is fatal. With this puzzle the player can keep trying until he/she gets it right. A bug I found here may be useful to other authors using portable containers that the player can enter. In this case the barrel. Initially the barrel was conveyable by the player.

So while carrying the barrel our hero could type ENTER BARREL and find themselves in the barrel! No problem, unless you type INVENTORY and find that the player is not only carrying the barrel but is in it as well!!!EEEKK!! My solution was to make the barrel to heavy for the player to carry. However should you want the container to remain conveyable then the following will do the job and avoid the above bug; Barrel is object 2,

ENTER BARREL PRESENT 2 CREATE 2 GOTO 2 DESC

PRESENT 2; Checks that barrel is here. CREATE 2; Moves the barrel to the current location. This has the same effect as dropping the barrel should the player be carrying it.

As you will probably know when creating puzzles you have to allow for the player trying things other than the right things. For the above simple puzzle I had to allow for; the player trying to knock on the door while in the barrel- Response "I just can't reach!"; the player not being in the barrel when the door is open, or being in the barrel but visible when Blotto opens the door- Response Wisecracks from Blotto!; The player trying to roll the barrel, (This is how Blotto gets the barrel into his house) Response-"Barrel rolling is a job for professionals, best leave it to them!"

Next time, I'll go into the preparing of the game for playtesting and hopefully news on the first report.

---

### PUZZLED

by Steve Clay

In an attempt to solicit articles for Coder, I thought I might explain how one of the puzzles in my game, The Taxman Cometh, (available from Zenobi and WoW software), came about, in the hope that other authors would follow suit.

The puzzle occurs in Phoebe's lair. Phoebe being the local dragon. You reach a flight of numbered stairs. If you attempt to go up them you are told they are out of order!

The object is to get the stairs back into the correct order. To do this you have to enter various rooms which do set things to the stairs. Room 1 may swap stair 1 and 3, room 2 may shift all the steps right, room 3 swaps stair 3 and 5, room 4 shifts all steps left and room 5 swaps steps 2 and 4.

This can cause the player serious headaches! One of the people who playtested the game attempted this puzzle 'blind' and quit 1600+ moves later. However when he thought about it he solved it in 3 moves which p\$%#\*&\* me off no end.

This puzzle came about one day when I was looking at how machine-code shift operations operated. So come all you genius writers out there, how did your puzzles get themselves created? What inspired you to put the key to the chest in the stomach of a yeti? Go on write in and tell the world.

# Whole Totta STYC

Just recently, Chris' sent me a letter from a guy named Kyoji Huchinawa, and always pleased to help.. I jumped in at the deep-end like an idiot!

Kyoji, you wanted some available exits printed, OK! But, did you want them on-screen ALL the time, did you want to be able to turn them on and off?

Did you want them printed underneath the location description at the top of the screen, or just printed with the rest of the messages at the bottom of the screen?

How about them being printed if you try and move in the wrong direction???

You didn't say, did you?

You didn't say what you wanted, or what you didn't want either!

Ab: well- we're put on this Earth to suffer aren't we?

So, if I can start by giving you a set of 'fixed' on-screen exits that are there all the time, with added on / off routines.

Now, these routines are developments / add-ons to some ideas given by Matt Conway in his 'Whole STAC of problems' column in 'Coder' issue 2, many moons ago! So, firstly, SOME messages...

3) , ,  
5) or  
9914) , ,  
(Remember those spaces!)

.. For the beginning of SPECIAL 17 'the start up process'. Matt' originally gave the full-stop as message 1, but why have 2 full-stops in a game? So, I've just used the one that's already built it. The rest of the message numbers are there to fit in with the rest of Matt's routines.. Now my additions to SPECIAL 17, below, simply set up Strings\$ 1, 3 and 5 to hold the text of messages 9914, 3 and 5, respectively..

9914 MESS\$ 1 LN MESS\$ 2 LN MESS\$ 5

In SPECIAL 13, which prints the prompt and 'status-line' at the top of the screen, you could also include something like:-

14) IF ( SET? 150 ) THEN MESSAGE 13 SPECIAL 20

With message 12 = 'HOME1DOWN1CUR1DOWN10by10H\_exit(-)

Which simply takes the cursor to the top of the screen, where the status-line (room name, score and room number) is printed, moves down 1 line, clears it so that nothing that moves underneath it, when the screen scrolls upwards, gets printed. The cursor then moves down another line to give a clear blank line between the status line and directions printed (makes them better to see!), it then prints message 12 and then goes onto SPECIAL 20 and lists any available exits.. So for SPECIAL 20 first we need some objects to use as direction names..

31) north (32) south (33) east (34) west (35) up (36) down  
37) northeast (38) northwest (39) southeast (40) southwest  
41) inside (42) outside

And the routines for SPECIAL 20

```

1) IF ( CONNECT "N" ) THEN 31 TO 9998
2) IF ( CONNECT "S" ) THEN 32 TO 9998
3) IF ( CONNECT "E" ) THEN 33 TO 9998
4) IF ( CONNECT "W" ) THEN 34 TO 9998
5) IF ( CONNECT "UP" ) THEN 35 TO 9998
6) IF ( CONNECT "DOWN" ) THEN 36 TO 9998
7) IF ( CONNECT "NE" ) THEN 37 TO 9998
8) IF ( CONNECT "NW" ) THEN 38 TO 9998
9) IF ( CONNECT "SE" ) THEN 39 TO 9998
10) IF ( CONNECT "SW" ) THEN 40 TO 9998
11) IF ( CONNECT "IN" ) THEN 41 TO 9998
12) IF ( AT 2 AND SET? 17 ) THEN 41 TO 9998
13) IF ( CONNECT "OUT" ) THEN 42 TO 9998
14) REPEAT
   ( ( FIRSTOBJ 9998 ) + 7970 ) MESS$ 0
15) IF ( CNTOBJ 9998 = 1 ) THEN 1 ADD$ 0
16) IF ( CNTOBJ 9998 = 2 ) THEN 5 ADD$ 0
17) IF ( CNTOBJ 9998 > 2 ) THEN 3 ADD$ 0
19) PRINT$ 0
21) FIRSTOBJ 9998 TO 0
22) UNTIL ZERO? FIRSTOBJ 9998
23) MESSAGE 40

```

So basically, what lines 1 - 11 & 13 do is, if there's an available exit in the current room, the routines move the object with the same direction name to room 9998. Now line 12 is special because if there's a 'conditional' exit that's only printed if something special happens (like a door to the west being opened SET? 17) in a specified room, then this takes care of it! Lines 15 - 18 take care of the printing of these exits. Line 15 tacks the appropriate message (1, 3 or 5) onto the end of each available exit-name, and lines 16 (prints full-stop), 17 (prints ' or ') and 18 (print',') figure out which message for line 15 to 'tack on'. The available exit objects are then sent back to un-created, and message 40 printed..

Message 40 = '[CR]' (Carriage return / New-line)

Now, in connection with that small routine in SPECIAL 17 there are these 3 routines in LOW Priority..

```

1) IF ( ZERO? NOUN1 AND VERB "Exits" ) THEN MESSAGE 12 SPECIAL 20
  NEWCOM
2) IF ( VERB "Exits" AND NOUN "On" AND RESET? 150 ) THEN CHANGE 150
  MESSAGE 9905 WAIT ELSE IF ( VERB "Exits" AND NOUN "On" AND SET?
  150 ) THEN MESSAGE 6 NEWCOM
3) IF ( VERB "Exits" AND NOUN "Off" AND SET? 150 ) THEN CHANGE 150
  MESSAGE 9905 WAIT ELSE IF ( VERB "Exits" AND NOUN "Off" AND RESET?
  150 ) THEN MESSAGE 6 NEWCOM

```

With message 12 as before. message 9905 = 'Okay!'

Message 6 = 'But, they already are!'

I've given 3 ideas here for you to choose from.

Now, line 1 just prints the exits in amongst the normal messages, after the prompt line. Line 2 turns the exits on so they appear permanently at the top of the screen, but if they're already 'On' then a message tells the player just that! Line 3 turns the exits off, if they are already printed at the top of the screen, but if they're already 'Off' then a message tells the player so!

As for your EXAM / SEARCH ideas and Get object out of / Put object into a container ideas, Kyoji, I'll have to think about them, so maybe next issue?

To start with even the cover of the cassette is quite eye-catching, black lettering on white, giving you a quick run-down of just what an issue might cover (P.D., Quill, GAC and PAW listings, letters, help given and reviews of games, utilities, etc) with a quaint seagull centre image to the cover!

When I loaded 'Adventure Link' (Part I of 9 loads!) into my old +2 (I loaded PAW in first, as 'Link' was written with PAW, and then loaded 'Link' as a file) I found, as with issue I that it worked out very well indeed, with facilities for 5 different fonts!

This tape-based magazine (issue 2) looks just like it wanted to be a game, with location descriptions and exits given (which lead onto other reviews, etc), so I played it like a game...

So, first up (after introductions) is the heading...

Directions:

NORTH = NEWS-LINK  
 SOUTH = IN THE PIPELINE  
 WEST = LETTERS  
 EAST = STAR LETTER

Where to now?

But firstly I tried the new fonts. FONT 1 is the font the tape starts off with. Not bad and easily readable. FONT 2 is an awful, hard to read Gothic/Olde English typeface. FONT 3 is a very slightly better, but still god-awful Gothic/Olde English typeface. FONT 4 is a very easy to read 'blocked' typeface. FONT 5 is a funny, but still readable Japanese/Asian-style typeface (or at least that's what it looks like to me!).

So do yourself a favour Alec (Carswell, the man who's put this overall wonderful Adventure Link tape-zine together), use 2 different fonts instead of fonts 2 and 3!

So I went back to FONT 1 for the rest of the review and carried on... I went NORTH to NEWS-LINK, which lists a company named 'Delbert The Hamster Software'. I'll call it DTHS from now on, set up by one Scott Denyer (more on him later), who, it says here, wrote a Conan the Barbarian spoof (or at least it sounds like one!) called 'Arnold The Adventurer' for John Wilson's Zenobi Software. NEWS-LINK also gives an overall view/review of 'Desmond & Gertrude', a new game by said Scott Denyer to be released on his DTHS label at £2.99... (Alas, the DTHS label has now ceased. - Ed)

Next is a list of some games released on Gordon Inglis' own GI Games label. Next up is a rather unusual company by the name of 3PSoft, from Spain! NEWS-LINK also mentions that of course all games will be in Spanish! Now, before you start moaning about not reading Spanish, the reason 3PSoft are in Link is because they're on the lookout for English games to translate into Spanish. So if you want to earn extra cash by selling overseas, write to...

3PSoft, APARTADO DE CORREOS 45076, 28080 MADRID, SPAIN.

Next mentioned is a company called 'The Guild', who, like GI Games, have also become a release company for oldies but goodies, like 'Puzzled', 'Into The Mystic' (?), 'Book Of The Dead', 'Deathbringer', from:

The Guild, 760 TYBURN ROAD, ERDINGTON, BIRMINGHAM, B24 9NX.

New fanzines are next listed by NEWS-LINK, starting with a new tape-based magazine named 'Alch-News', created by the aptly named 'Alchemist Research', who also supply FREE fonts, and character sets. With an SAE sent to...

Andy Davis, Alch-News, Alchemist Research, 62 TITHE BARN LANE, WOODHOUSE, SHEFFIELD, S13 7LN.

Next is Tim Kemp's (name rings a bell!) FBPD Library listing several games.

With new releases from Compass Software, 'The Antilles Mission', 'Blood Of Bogmole II'. (Sic(k) - Ed.) Zenobi... 'T'Was A Time Of Bread' (the first rasta adventure? - Ed.), 'The Curse Of Calutha', 'Violator Of Voodoo', 'Arnold II', 'Jester's Jaunt' etc. Larry Horsfield's Fantasy Science Fiction (PSF) Adventures... 'Run, Bronwynn, Run', 'Spectre Of Coris Castle'. (Ed - I've had this review for a while now, so apologies if these are old games by now!)

NEWS-LINK also offers free software to the person who offers the most contributions to each issue! A good deal if ever I heard one!

Back at the main heading, I went SOUTH to IN THE PIPELINE, giving a review of Compass Software's new game, 'The Antilles Mission'.

WEST led to the LETTERS pages listing 8 huge letters from various people, but you're going to have to buy Adventure Link to find out who!

EAST leads to the STAR LETTER and huge it is too!

Load 2 of Link gives the heading...

NORTH = FANZINE FRONT  
SOUTH = EXTERN EXPRESS  
WEST = FANZINE FOCUS  
EAST = PAWS 'N' CLAWS

So I went NORTH to FANZINE FRONT, listing every 'known' fanzine (tape and paper-based). It next lists a lovely, detailed run-down of some of these fanzines, giving price, a list of content of the issue reviewed and current issue number as well.

Next was SOUTH to EXTERN EXPRESS, giving a nice, little surprise EXTERN routine, but you're going to have to buy Adventure Link and try the routine to find out!

WEST went to FANZINE FOCUS, giving details for 'Red-Herring' fanzine. Which is a hell of a read through, but very good!

EAST led to PAWS 'N' CLAWS, giving a nice little 'LAST' routine for PAW, from Mark Cantrell.

Load 3 gives the heading...

NORTH = UNDER PRESSURE  
NW = FRIENDLY SPIRITS  
NE = DOCTOR DOCTOR  
SOUTH = SLIMY SECRETS  
SW = FIND THAT OBJECT  
SE = BEWILDERING SPIRITS  
WEST = TRANSTIPTIONS (?)  
EAST = SOLUTIONS

NORTH I went to UNDER PRESSURE, listing a specific older game which people may be having trouble with.

FRIENDLY SPIRITS (hic!) gives a request for people who've solved a game to send in their answers to help others!

DOCTOR DOCTOR is a service for people who can't find something, such a game, utility, etc, that they're after!

SLIMY SECRETS is a service giving secret passwords, cheats, hacks, bugs found, etc, such as some weird names giving weird effects in 'Behind Closed Doors'.

FIND THAT OBJECT gives a complete list of where to find the hidden objects in Linda (Marlin Games) Wright's old smashers, 'Cloud 99' and the 'Jade Stone', now released by Zenobi. FIND THAT OBJECT also has lists for finding objects in John Wilson's 'Balrog & The Cat', and 'Urban Upstart'.

SE led to BEWILDERING SPIRITS (I've drunk a few of them in my time!). This is a

section for those who are well and truly stuck and don't know what to do next.

WEST is TRANSTIPTIONS (whatever it means) which is a very good hints and tips section.

EAST was SOLUTIONS, giving complete solutions to 3 games.

Load 4 carries the heading...

NORTH = PRACTICAL PROGRAMS  
SOUTH = POPULAR POKEs  
WEST = PD LISTS  
EAST = PD CONNECTIONS

PRACTICAL PROGRAMS gives programs to add onto your game loaders for border effects, etc.

POPULAR POKEs likewise gives pokes to make your programs just that little bit better!

PD LISTS gives just that, a list of various PD utilities... Viewdata, fonts, emulators, printer-drivers, hyper-load facilities, screen-dumps, inlay-card printer programs, etc, etc, etc. All from...

Propac Computer Group, Propac PD, I53 PICKHURST RISE, WEST WICKHAM, KENT, BR4 0AG.

Or, Adventure Link has its own PD programs... Multiflash loading screens, screen printer, border FX, etc.

Load 5 has the heading...

NORTH = CONTRIBUTIONS CORNER  
SOUTH = NEXT ISSUE  
WEST = BACK ISSUES  
EAST = SUBSCRIPTIONS

CONTRIBUTIONS CORNER is a list of all those who've given help in any way to the creation of the current issue of Adventure Link.

NEXT ISSUE tells you just what's coming up in the next issue, and is bloody ginormous by the look of it!

BACK ISSUES gives a list of past copies. Only one issue so far!

SUBSCRIPTIONS gives a list of joining prices for Adventure Link, with a list of freebies to choose from when getting a subscription.

Load 6 is a totally separate program to Adventure Link called Alch-News, which is a teletext lookalike. It starts at page 100 like the real teletext on TV does, with pages thus: 100-109 are blank; 110-113 telling you just what Alch-News is; 114-119 are blank; 120-121 are pages asking for contributions; 122-129 are blank; 130-133 are for if you have a problem but not the know-how to fix it, or you have the know-how but not the problems to solve for people! Pages 134-139 are blank; 140-151 are all exactly the same, and are help areas for just about every machine going (except the ZX81!); 152-160 are likewise all exactly the same, showing what's in the next issue of Alch-News, not Adventure Link! Page 161 is a second page giving more of what's in the next issue; 162-163 are both the same, showing a third page of what's in the next issue of Alch-News, and asks for any help, hints or adverts, etc, people want to put in; 164-170 are all exactly the same, and for some reason, all they show are 5 crosses 'XXXX' at the top of the screen! Pages 171-180 are all exactly the same, showing a list of adverts appearing on pages 181-187. Page 181 is an advert, with address for 'Z.A.T.' fanzine (also on page 193 for some reason?); 182 is an advert for the Romantic Robot Multiface; 183 is an advert for Alchemist Research (also on page 191 for some reason?); 184 is an advert for Domark; 185 is an advert, with phone numbers, no addresses, for SAM Coupé Hardware and the SAM User Group; 186 is an advert for Datel Electronics; 187 is an advert for Adventure Link; 188-189 are blank; 190 gives a list of useful contacts; 191 is an advert for Alchemist Research; 192 is an advert for Off-Tel Bulletin Board Services; 193 is an advert for Z.A.T. (whoever they are?);

I94 is an advert for Andy Glaves Software (who?); I95 is an advert for Brian Gaff Services (who?); I96 is an advert for User Friendly Laboratories Ltd. (who?) (Ed - I think those last three might be in-jokes and not real!) Page I97 is an advert for Steve Shephard (also known as DEBUG), but there's no address; I98 is an advert for Incentive Software; I99 is the last readable page, and lists the final credits for Alch-News; 200-201 - these pages are still accessable but are blank.

Load 7 is a nice demo of a 16-bit machine Windows-Icons-Mouse-Pull-down Menus (WIMP) system lookalike!

Load 8 is a lovely Spectrum Monitor allowing memory to be examined, pokes entered, saved, loaded and a Checksum made. A lovely little freebie!

The last piece on the tape, Load 9, is a marvelous little Sound Sampler, which really does work, which records, plays back and allows you to fiddle with human voices, music, dog barks, etc, easily. I've tried!

For more information about Adventure Link, write to...

Alec Carswell, 16 MONTGOMERY AVENUE, BEITH, AYRSHIRE, SCOTLAND, KA15 1EL.

#### THE ADVENTURE COOLER AWARDS FOR... BEST ALBUMS OF 1992

- 1.) 'Hypocrisy Is The Greatest Luxury' - The Disposable Heroes Of Hiphoprisy
- 1.) 'Automatic For The People' - R.E.M.
- 1.) 'Broken' - Nine Inch Nails
- 4.) 'Dream' - Kitaro
- 5.) 'Amused To Death' - Roger Waters
- 6.) 'Greatest Misses' - Public Enemy
- 7.) 'Fixed' - Nine Inch Nails
- 8.) 'Astronauts & Heretics' - Thomas Dolby
- 9.) '3 Years, 5 Months And 2 Days In The Life Of...' - Arrested Development
- 10.) 'The Predator' - Ice Cube

#### BEST SINGLES OF 1992

- 1.) 'I Can't Dance' - Genesis
- 2.) 'Constant Craving' - kd Lang
- 3.) 'Hazard' - Richard Marx
- 4.) 'November Rain' - Guns N' Roses
- 5.) 'Money Don't Matter 2 Night' - Prince
- 6.) 'Layla (acoustic)' - Eric Clapton
- 7.) 'One' - U2
- 8.) 'Shake Your Head' - Was (Not Was)
- 9.) 'Midlife Crisis' - Faith No More
- 10.) 'Lithium' - Nirvana

COMMISSIONERS TO... Simply Red, R.E.M., Sir Mix-A-Lot, House Of Pain, Kris Kross, Naughty By Nature, Public Enemy, Thomas Dolby, Arrested Development, L7, Fish, Shakespears Sister (their spelling, not mine!), Prefab Sprout, Madonna, and more.

#### BEST MAGAZINES OF 1992

- 1.) 'ST Format' - Future
- 2.) 'Hip-Hop Connection' - Popular
- 3.) 'Atari ST Review' - EMAP Images
- 4.) 'VOX' - IPC
- 5.) 'The Amazing Pudding' - Mabbett, MacDonald, Walker & Walker
- 6.) 'YES Music Circle' - Tiz Hay
- 7.) 'Goblin Gazette' - Les Mitchell
- 8.) 'Select' - ?

To follow on from the John R. Barnsley's PD Library review I did for 'Adventure Coder', I have here a short list of 'Adventure Writing Utilities'. prices and the addresses they may be bought from, all in PD Library form.. But a couple of small points first..

A) Please understand that in the case of any PD disks, because they 'are' PD (or Shareware / Licenseware, etc) it's entirely possible that any 'Adventure Writing Language' may exist in the exact same form for a number of different PD Libraries, but under *different names*, ie, it IS possible to send off for 2 separate creators, with 2 different names, from 2 different companies, and get *exactly the same package* back in both cases, so, you have been warned!

B) I'm also giving a warning to any newer ST PD adventurers, about the horrors of the multitude of Virus' (viruses?) which abound in the Public Domain, including the awful 'Letter' or 'Text-Melt' virus I've encountered twice from 2 entirely different PD Libraries! Neither of which I've included in the short listings here! Amongst the PD 'Adventure Writing Utilities' I have included here, are..

Code.	Name.	Price in pounds.
LAM.3	The Adventure Writer	1.50

For the ST / STE on a Single/Sided disk, from..

L.A.P.D., 80 Lee Lane, Langley, Henor, Derbyshire, DE7 7HN  
Tele: 0773 - 761944 / 0773 - 605010

L.A.P.D. also do a Single/Sided STACed text adventure (for 1.50) called 'Don't Bank on it!' written by Ruth Sunderland (code LAG.114). Ruth's had letters printed in issues 9 and 10 of 'Adventure Coder', in which she asked an important question for many adventure writers, see the end of this column for an answer to Ruth's question from issue 10!

LAN.4	Adventure Writing System	2.00
-------	--------------------------	------

Presumably on a Single/Sided disk, from..

Softville Computer Supplies, Unit 5, Stratfield Park, Elettra Avenue,  
Waterloooville, Hants, PO7 7PY  
Tele: 0705 - 266509

L.3	ADVSYS; ADventure writing System	3.00
-----	-------------------------------------	------

Reviewed by myself in 'Adventure Coder' issue 11. It also comes with a mediocre adventure called 'Starship Columbus' (but PLEASE read the review before buying!), on a Single/Sided disk, from..

Public Dominator ATARI, PO Box 801, Bishop's Stortford, Herts,  
CM23 3TZ

Public Dominator ATARI also do an 'Adventure Writing System' for three pound, called the AGT (Avenue ams Toolkit), version 1.19, used to write many games for John Barnsiev's PD Library, I'll mention him later!

ST.114 ADVSYS; ADVenture writing SYSTEM 2.00

See L.3 above, available from..

Gemini Software, 10 Warwick Avenue, Slough, SL2 1DX  
Tele: 0753 - 28183

F.005 Adventure Writing System 2.00?

Available from..

M. T. Software, Greens Ward House, The Broadway, Totland, I. O. W.,  
PO39 0BX  
Tele: 0983 - 756056

no' 352 The Eamon Writing System 1.49

More of this utility later on, available from..

Computing Connections, Ashlaw House, Euxmoor Drive, Christchurch,  
Wisbech, Cambridgeshire, PE14 9LS

Computing Connections also do disk no' 309 'The Adventure Writing System' for 1.49, but as it says that this system also includes the adventure 'Starship Columbus', this is obviously in reality L.3 'ADSYS'!

The 'Eamon Writing System' is also sold by..

Akore, Nottingham, NG1 1BR

Under the title..

GA.6204 The World of Eamon 2.49 = 1.95 P&P

Akore also do..

GA.6314 AGT Adventures 2.49 = 1.95 P&P

Which is really the AGT Source Code, see SW.8, above.. Also listed in their catalogue is something for the Amiga called..

GA.7199 Amiga-venture 2.49 = 1.95 P&P

Which Chris' mentions in his 'Utilities and Add-ons' column in 'Adventure Coder' issues 2, 3 and 4. Akore also does an Amiga version of the AVDSYS..

UT.4282 ADVSYS version 1.2 2.49 = 1.95 P&P

ADV.06 ADVSYS: ADVenture writing SYStem 2.50

See L.3 above, is also available from..

Titan Games, 45 Windmill Lane, Worksop, Nottinghamshire, S80 2SO.

**ADVSYS** and **Starship Columbus** are also available on a compilation disk, 'Double Adventure Disk.3' from Titan Games, along with 4 text adventures, 'Once a King' (written by Darryn Lavery and Mark Griffiths, names ring a bell from somewhere?), 'Crowley's House', 'Prisoner of the Dark Pearl' and a Gordello Incident look-alike, 'Dappyworld' (which is about the best of the 4. It's the only decent 1 of the lot, really!).

SW.7           AGT; The Adventure Games Toolkit           1.50

This is Shareware, so if you really do 'get into' this program, in a writing way, you should donate something to its original writers! Available from John R. Barnsley, address at end of column. John's PD Library also does..

ADV.36           ADVSYS; ADVenture writing SYStem           1.50

Complete with Starship Columbus, see L.3 above..

SW.8           The AGT Source Code           1.50

For SW.7 above, self explanatory really? And..

ADV.27           ST Adventure Writing System           1.50

Another adventure creator, written in GFA Basic 2, by R. D. Foord.

Now I really don't know how good any of the adventure creators above are, except in the case of ADVSYS from Titan Games, but if you want to get yourself a CHEEP, readily available, with plenty of back-up support and help from their writers, adventure creator, you might as well give one of them a go!

Now to end with, I'll try and answer Ruth Sunderland's question about how to sell any self written (in her case STAC written) adventures via the Public Domain. Though she seems to have solved that problem for herself quite admirably, as you can see by LAG.114 'Don't Bank on it!', above! I've encountered enough PD Libraries by now to know that 1 of (if not THE!) best 'Adventurer's' PD Libraries so far is..

John R. Barnsley's ST Adventurer's PD Library, 32 Merrivale Road, Rising Brook, Stafford, Staffordshire, ST17 9EB

Now, to answer Ruth's question, on the front page of John's A4 size, paper-based PD catalogue (his disk-based version is far, far bigger!) it specifically says that if you've written an adventure which you wish to enter into the Public Domain (STAC, Machine Code, Basic or whatever), he'll be more than happy to receive and assess any such item with the view of simply 'exchanging' your 1. own that you personally have written game, for 3 that he already has on PD (I can personally recommend BU.29 'Death Camp' and BU.43 'The Village', a couple of text adventures by our own Dave Blower! (From Budgie Shareware!) BU.68 'Blackscar Mountain' also Shareware, by Terry White, or, COM.19 a compilation disk with both 'Death Camp' and 'The Village', or COM.21 a second compilation disk with both 'Blackscar Mountain AND Shards of Time' ('Adventure Probe' readers will know of the advert' for these 2!). There's also ADV.15 which might be worth you while, a Bard's

Tale-task RPG system called the 'Eamon Adventure System', and lastly I can recommend SW.2 'The Grimoire', a STACed adventure by Richard Lee, and SW.3 'Toil and Trouble' a STACed adventure by once ACE columnist, 'Adventure Contact' writer and 'Adventure Probe' co-founder, Pat Winstanley!). Meaning that, after all that, you won't get paid in money, but you will get paid in kind! Now onto Licenseware and Shareware PD, unfortunately I have no addresses for these but somebody, possibly Ruth herself, might have some help to offer! Is there anyone listening?

There's so many more different PD 'Creators' I haven't been able to track down, like the PD version of ADL, 'ADventure Language', for the Amiga, but if anybody has any more PD 'Creator's to add to this list, why not write in to Chris', or myself at..

8 Heatherslaw Road, Fenham, NEWCASTLE UPON TYNE, NES 2QJ

# STOP PRESS!!

LES MITCHELL  
JUST SENT  
ME THIS...

It is with regret that I have to inform you that, although I have thoroughly enjoyed producing Goblin Gazzette, I am now in the position were I can no longer continue to subsidise the production costs of the fanzine.

I will be perfectly honest and tell you that in total I have 41 loyal subscribers. It does not take a great mathematician to work out that the fanzine is running at a loss. I have for the last two issue been able to meet this loss out of my own pocket with the hope that subscriptions would pick up, however, this proved not to be the case.

I would like to personally thank each and everyone of you who subscribed to the fanzine from day 1 and helped it to mature to what I thought was a damn good read, and to something which was on par with current fanzine's. I suppose I could go on and tell you why I think the subscriptions failed to pick up, and for the apparent lack of interest, but that would be crying over spilt milk and like they say 'It's better to have tried and failed than to have not tried at all'.

# STOP PRESS AGAIN!!

THE ABOVE - THIS CAME A FEW DAYS LATER...!  
IGNORE

You recently received news regarding the demise of Goblin Gazzette which may or not saddened you. I know that this will come as a bit of a surprise to you, but I am now in a position where I can continue to produce Goblin Gazzette even with it's small number of subscribers.

The reason for this is I have managed to make a deal with a local printer who just happens to be a friend of mine, who is happy to reproduce G.G. at half the costs that I was able to produce it.

GOBLIN GAZZETTE  
10 TAVISTOCK STREET NEWLAND AVENUE HULL HU5 2LJ

How to Make a Thingy (Part IV)  
by Stephen Groves

LOCPRN

We are now about to continue with LOCPRN and it's associated routines. As stated earlier, LOCPRN is the main printing routine which handles both the position of text on the screen and wordwrap. It also holds the library for compressed text and handles compressed text expansion before printing.

The main entry point is at the label LOCPRN, but there is another entry point at NXTLIN which will start printing on a new line if required.

WORDWRAP

In considering how wordwrap works, we must give some thought to what a word is made up from. A sentence, is formed by groups of letters placed together. Each group of letters forms a separate word and is separated from other words by either a space or punctuation. Generally speaking, unless a word is the last word to be written in a paragraph, punctuation is also separated from the next word by a space. It seems logical therefore to look upon a space as being the terminator of any word. When using computers, we also have to have a separate terminator that shows when a section of text is complete. For this terminator, I am using the number 13d (0DH) which signifies the 'carriage return'. We now have two distinct numbers that show where a word ends. ie.

1. The space = ASCII code 32d (20H)
2. The carriage return = ASCII code 13d (0DH)

In LOCPRN, we take the column number showing where the next character is to be printed, from COLPOS. We increase this position and re-store it in COLPOS after checking that it is still on the same screen line. We then get the next letter to be printed and check to see if it is a printable character or a word terminator.

If it is a character, we loop back and carry out the same process again. If we come to the end of a screen line before we come to the end of a word, a jump is made to NXTLIN, and printing is carried out on the next screen line.

If however we find the next character is a word terminator, then we already know that this word will fit onto the current line and we go ahead and print it.

It may be worth noting, that I do not consider punctuation to be a word terminator. It is considered to be part of a word. Imagine a sentence, where the last word fits snugly onto the end of the line. If punctuation were a word terminator, then it would be considered as separate to the last word and the punctuation would be printed as the first character of the next line. I think you'll agree it would look decidedly odd.

Finally, on the subject of wordwrap, if you come across a word 33 or more letters long, DON'T USE IT WITH THIS PROGRAMME. The computer will continuously move to the next line, trying in vain to find a line it will fit onto. All you get is a screen full of nothing, with the word "more..." at the bottom. (Can you really think of a word with 33 letters that would be needed in a adventure?)

### TEXT COMPRESSION

Text compression, is a method of saving memory. The normal Spectrum character set has ASCII codes in the range 32d to 127d. Numbers outside this range are either un-printable or are peculiar to the Spectrum (ie KEYWORDS).

By looking carefully at any section of text, you will see that some letter groupings and even some words are repeated more than others. In an adventure you very often find groupings such as :

```
"You "
"the "
"ing "
"are "
"el"
". "
```

These are just a few of the many you can find. Now, if we replace one of these letter groupings with a single number that is not between 32d and 127d, we can save memory space. The replacement numbers I have chosen are all greater than 0A4H. Obviously we need a library of some sort, so that we know which number refers to which group of letters. The library is 405 bytes long and starts at COMDAT.

The library that I have used, achieves a saving of between 35-40% of memory space. This means that if you have 20K of memory available for text, you can squeeze about 27 to 28k of text into it.

I used a complicated mathematical process to decide which letter groupings to compress, it is called TRIAL and ERROR. If you think that some groupings are not necessary, then feel free to replace them with your own.

In the listing of COMDAT, all spaces in the letter groupings will be shown as "\*". So if you see "all\*" you will know it is the word "all" followed by a space.

### SCROLL?

The decision whether to scroll the screen comes under the control of NXTLIN.

In both graphics and text mode, printing takes place down to and including line 22. When the screen is full of text, the word "more..." appears on the bottom left of the screen and the programme stops while waiting a key press. Following the keypress, the screen can then be scrolled one line either 13 times (graphics mode) or 23 times (text mode). The counter for the number of times one line can be scrolled is kept in SCRCT. Following a scroll, printing resumes on line 22.

The actual scrolling is carried out by the ROM routine that resides at 0E00H.

#### CHARACTER SET

The Spectrum character set is pointed to by the lable CHSET. Actually, CHSET points to 256 bytes less than the start of the character set in ROM. The reasoning behind this is quite simple.

The first printable ASCII character is the space, which has ASCII code 32d. Each character is made up of 8 bytes. It just so happens that  $8 \times 32 = 256$ , so if you multiply any ASCII code by 8 and add it to CHSET (256 below the start of the characters), you will end up pointing at the first byte of that character.

Should you wish to design your own character set, CHSET should be made to point to 256 less than the memory address where your character set is located.

I hope that I have covered all the major points relating to the printing routines, any other points will be covered in the comment lines accompanying the listing.

CHSET	EQU 3C00H	;256 less than the Spectrum character set
NXTLIN	XOR A	;Load A with zero...
	LD (COLPOS),A	;...and make it the current COLPOS.
	LD A,(LINPOS)	;Load A with the line No.
	CP 16H	;and compare it with the last line on the page.
	JP Z,OLDDLIN	;If it is the last line we need to scroll.
NEWLIN	INC A	;If it isn't the last line point to the next line
	LD (LINPOS),A	;and store it.
	BIT 0,(IX+14H)	;Test a flag used by INPUT to see if a return is required.
	RET NZ	;Return if flag set by input.
	JP LOCPRN	;Now jump to LOCPRN to print on a new line.
OLDDLIN	LD A,(SCRCT)	;Place the number of times the screen can be scrolled by one line in A
	INC A	;Check if A contains zero
	DEC A	
	CALL Z,MORE	;If it does, print MORE..
	DEC A	;Decrease the number of scroll times by one

# AMSTRAD ADVENTURE P.D.

*Don't let your adventures collect dust - Get your work seen!*

A newly formed PD specialising in adventures only is seeking home-brew adventures to include in the collection. Send your tape or disc to:

ADVENTURE PD, 10 OVERTON ROAD, ABBEY WOOD, LONDON, SE2 9SD

*For more information send a SAE to the above address.*

```

LD (SCRCT),A ;and store it.

CALL SCROLL ;Scroll one line

LD A,16H ;We want to print on line
          ;22d
LD (LINPOS),A ;and store it.

BIT 0,(IX+14H) ;Another flag test to
                 ;see if a return is
RET NZ        ;required by INPUT

;The main printing entry point, and a continuation from
;OLDLIN, unless OLDLIN was called from within INPUT.

LOCPRN      LD HL,(LOCPOS) ;The address of the text
              ;to be printed
              LD A,(COLPOS) ;The current column No.

              LD (T.COL),A ;Temporary store

LLENTH       LD A,(T.COL) ;Retrieve the current
              ;column No....
              INC A        ;and increase it for the
              ;next character...
              CP 22H        ;to see if we are past the
              ;end of the current line
              LD (T.COL),A ;Save the column No again

              JP Z,NXTLIN ;Jump to NXTLINC if we
              ;have exceeded the line
              ;length
              LD A,(HL)    ;If not, we load the next
              ;character into A
              CP OFFH      ;Check if it is the
              ;terminator at the end
              ;of compression..
              CALL Z,COMP2 ;Call COMP2 if it is

              CP 0A5H      ;Check if it is the start
              ;of compressed text
              CALL NC,COMP1 ;Call COMP1 if it is

              CP 20H        ;Check if it is the end
              ;of a word (SPACE)...
              JP Z,CHKIT1 ;...and jump if it is

              CP ODH        ;Check if it is the end of
              ;text...
              JP Z,CHKIT2 ;... and jump if it is

              INC HL       ;If not, point to the next
              ;character in text...
              JP LLENTH   ;...and repeat

CHKIT2       LD HL,T.COL ;At the end of a piece of
              ;text, we don't print CARR
              ;RET (ODH) so...
              DEC (HL)    ;...we can discard it by
              ;reducing the word length
              LD HL,COLPOS ;Having found the end of a
              ;word (SPACE), we now find
              ;the word length by
              ;placing the original
              ;column pos'n in (HL)..
```

	LD A, (T.COL)	;...and the column pos'n at the end of the word in A
	SUB (HL)	;and subtract, leaving the word length in A
	PUSH AF	;Save the word length on the stack
	CALI DISAD1	;Construct the screen RAM pos'n from the current 'COLPOS'
	POP BC	;Retrieve the word length into BC
PRNTIT	LD HL, (LOCPOS)	;Place the address of the beginning of text in HL
	LD A, (HL)	;Load A with the current character...
	CP OFFH	;...and check if it is a terminator for compressed text
	CALL Z,COMP2	;Call COMP2 if it is
	CF 0ASH	;Check if if is compressed text...
	CALL NC,COMP1	;...call COMP1 if it is
	PUSH HL	;Save the text address on the stack
	PUSH BC	;Save the word length on the stack
	CALL PRINT	;Print the character
	CALL UPDATE	;Calculate the next screen RAM address for printing
	POP BC	;Restore the word length
	LD HL, COLPOS	; (HL) now contains the last column position..
	INC (HL)	;..so increase it to point to the next column pos'n
	POF HL	;Restore the text address from the stack
	DJNZ PRNTIT	;If the word length is >0 jump to PRNTIT to repeat
	LD A, (HL)	;At the end of a word, load A with the next character
	CP OFFH	;Check if it is the terminator of text compression
	CALL Z,COMP2	;Call COMP2 if it is
	LD (LOCPOS), HL	;Save the text address
	LD A, (HL)	;Place the next character in A
	CP ODH	;Check to see if is the end of text terminator..
	RET Z	;...and return if it is
	JF LOCPRN	;Otherwise, jump to LOCPRN to deal with the next word
COMP1	LD (T.POS), HL	;Save the text address
	LD HL, COMDAT	;Load the start of the

		compressed text data into HL
	PUSH BC	;Save the word length on the stack
	LD BC 0200H	;The length of the table (and a few to spare)
	CPIR	;Register A contains the compressed text token, and a comparison is made until a match is found
	POP BC	;When the match is found restore the word length
	LD A,(HL)	;Load the de-compressed character into A...
	RET	;..and return with it to either count or print
COMP2	LD HL,(T.LPOS)	;Restore the text address
	INC HL	;Point at the next character
	LD A,(HL)	;Move the character into A
	RET	;..and return with it to either count or print
DISAD1	LD BC,(COLPOS)	;B=line No. C=column No.
DISAD2	LD A,B	;Transfer line No. to A
	PUSH AF	;Save it on the stack
	CP 0	;Is it line zero
	JP Z,NO.B	;Jump if it is
	LD A,C	;If not, load the column No. into A
TWICEA	ADD 20H	;For every line down the screen, add 20H (the line length)
	DJNZ TWICEA	;Loop back if more lines
	LD L,A	;When no more lines left, store A into L
	POP AF	;Restore line No. to A
	CP 08H	;Is the line in the top 1/3rd of the screen
	JP NC,FORTY8	;Jump if not
	LD H,40H	;Load H with 40H if it is
	JP DISAD3	
FORTY8	CP 10H	;Is it in the middle 1/3rd of the screen
	JP NC,FIFTY	;Jump if not
	LD H,48H	;Load H with 48H if it is
	JP DISAD3	
FIFTY	LD H,50H	;It must then be in the bottom 1/3rd of the screen. So load H with 50H

DISAD0	LD (SCRPOS),HL	;HL now contains the screen RAM address, so store it...
	RET	;...and return.
NO.B	LD L,C	;On line zero, the low byte is the same as the column No....
	LD H,40H	;...and as line zero is in the top 1/3rd of the screen, we load H with 40H
	POP AF	;Remove AF from the stack
	JP DISAD0	;To return
PRINT	LD L,A	;Register A holds the ASCII code of the character to be printed
	LD H,00H	;HL now contains the ASCII code
	LD DE,CHSET	;DE now contains the base address of the Spectrum character set in ROM
	ADD HL,HL	;Multiply the ASCII code by two..
	ADD HL,HL	;...and by four..
	ADD HL,HL	;...and by eight
	RET	;Return when eight bytes printed
UPDATE	LD HL,(SCRPOS)	;(HL) now points at the current low byte of the screen RAM address
	INC (HL)	;Increase it
	RET NZ	;If the low byte doesn't become 00H then return
	INC HL	;If it does become 00H, then we have moved to a new 1/3rd of the screen..
	LD A,(HL)	;...so place the high byte into A..
	ADD 08H	;... and add 08H (eg. 4000H becomes 4800H)
	LD (HL),A	;Place the new screen RAM address back in SCRPOS..
	RET	;...and return
SCRPOS	DEFW C	;Storage for the screen RAM address
T.LPOS	DEFW 0	;Temporary store for LOCPOS
	ADD HL,DE	;Add to CHSET leaving the ROM address of the character in HL
	LD DE,(SCRPOS)	;Place the screen RAM address in DE
PRINT1	LD B,08H	;Use B as a counter for the eight bytes
AGAIN	LD A,(HL)	;Load A with the byte to be printed, held in (HL)
	LD (DE),A	;Transfer it to screen RAM (DE)
	INC HL	;Point at the next byte of the character to be printed
	INC D	;Point at the next byte of screen RAM..eg. 4020H becomes 4120H
	DJNZ AGAIN	;Loop back if BX0

# S.T.A.C. Animation!

by  
Dave Blower

Yes it is possible to animate with S.T.A.C. but this information will not be found anywhere within the manual (surprise, surprise!). Here is how it is done...once you have mastered the basic formula you can then 'let your imagination go berserk'. It is all a matter of switching top colours off and on, using the pause command and the right R.G.B. (colour settings). You can create the illusion of ; a gun firing to an apple disappearing off a tree coupled with the 'get apple from tree' type of command. I am going to give you a very basic example of a Castle door opening when you approach, then closing when you try to enter the Castle. First you must draw a Castle door with surrounding walls. I use Deluxe Paint but any art package that save in NEO or DEGAS formats will suffice. The colour palette consists of 16 colours do NOT use either 0 (the first colour of your palette) or 3, 0 will affect your 'screen colour' if you animate it and 3 will alter the colour of the Short Bar Description that runs along the top of your screen, use any other colours apart from these. We are going to use colours 4, 5 and 6 these are counted from the left of your palette colours.

		No 4 fill colour 333		
		No 5 fill colour 333		
		No 6 fill colour 333		

Draw the diagram, then fill the number 4 part of the door with R.G.B. colour 333, number 5 with 333 and finally number 6 with 333, ignore the *faint lines* that I have drawn that separates the fill colours, the fill colours must *touch* each other. Don't forget to use other palette colours for the rest of the drawing. Now for the programming...

The Castle door is at location 21 then from location 20 you would have to move north in order to get from 20 to 21. So create both the 'rooms' and import the same Castle pictures for both rooms.

#### Local Condition 20

**IF VERB 'N' THEN GOTO 21 LF MESSAGE 215 SPECIAL 201 LF WAIT**

**Message 215:- As you approach the Castle the ancient heavy door slowly rises.**

**Special 201:- 6 topcol 0 pause 30 5 topcol 0 pause 30 4 topcol 0**

This turns the palette colours from 333 to 0 (black) starting from the bottom, thus giving the illusion that the door is opening!

If you were standing on a pressure pad that 'raised' the Castle door as your weight triggered some hidden mechanism, then you tried to enter the Castle by typing in N your weight would move off the pad and the door would come crashing down, so this condition would apply.

#### Local Condition 21

**IF VERB 'N' THEN LF MESSAGE 211 SPECIAL 203 LF WAIT**

**Message 211:- The Castle door crashes down as you try to enter.**

#### Special 203

**4 topcol 333 pause 10 5 topcol 333 pause 10 6 topcol 333**

This creates the impression that the door has closed, by reversing the sequence and changing the three segments of the doors colours to 333. Notice that the pauses have been reduced to 10, this makes the door close quicker to marry up with the message 'The door crashes down!'

# BIRCHWOOD

## *Prisoners produce computer game*



Hoping their efforts will boost the Scanner Appeal are members of "Prime Time," from the left, Derek Lamb; Bob Elliott; Alan Allcorn and Officer Dave Blower.

Photo. supplied.

**GHOSTLY** goings on are set to aid the sick and ill thanks to inmates at Her Majesty's Prison, Risley.

Warrington District General Hospital's Treadmill Appeal is to benefit from the profits of a hi-tech computer game — "The Phantom Zone" — produced by prisoners.

The group, known as "Prime Time," was formed by prison officer Dave Blower in January, 1991, and is having its game marketed by Essex-based international software distributors, Budgie UK Licenceware.

Mr. Blower said: "Creating computer games for the Atari ST system has been

my hobby for 10 years, and I thought it would be a great idea to form a class at the prison.

"When we finished the programme, it was unanimously decided to donate all profits to the Scanner Appeal.

"We are now working on a second game which should be ready this time next year."

Anyone interested in buying a copy of the game should send a formatted 3.5 disk and a self addressed envelope, plus a cheque for £2.00 made payable to the Scanner Appeal, to "A" Wing Computer Class, Officer D. Blower, HMP Risley, Warrington Road, Risley, Warrington, Cheshire, WA3 6BP.

## AMOS AVENUE

by

By John Ferris

The previous four articles have run through the programming groundwork required for writing an adventure game in AMOS. Now we are going to put the programming skills together with the creative skills and write an adventure.

### The Plot

Included in this article is a simple map of the adventure. As you can see, this game isn't quite as large as Snowball but it will serve our purpose. I thought up the opening problem one lunchtime at work as I sat ruminating over my lunchbox. Imagine an apprentice wizard wandering through some ruins on some errand for his master when he falls through the roof of an underground chamber. This is effectively a stone pit and the poor apprentice is stuck in it. There are two possible ways out of the pit. The first is up through the opening in the roof which is strangely regular in shape. The opening is obviously too high to reach. The second way is by finding the secret door in the east wall. The latter was better as it opened up the underground exploration theme. Yes, this scores -7 on the originality scale. I remembered a scene from Raiders of the Lost Arc where Indy finds the correct location of the Arc by focusing light through a ruby on a model of the ancient city. So, the apprentice will be carrying a staff with a crystal on the top, and the floor under the opening will have a hole in which to stick the staff. The sunlight streaming through the hole in the roof will be focussed by the crystal and operate a light sensitive lock and the secret door will open.

This is where I get sneaky. First, there's only eight moves worth of sunlight before the sun is out of position. Second, once the staff is removed from the hole, as it needs to be, the door shuts unless it's wedged open with a handy rock. The staff is needed as it doubles as a light source needed after location 3. Yes, your friend and mine, the GRUE! happens to be lurking nearby. A magic word (inscribed on the staff) will make light.

I shall now score -10 on the originality scale by stealing an idea from the original adventure, that of the plant which grows when watered. Locked in the storeroom, location 4, is a magic seed. You know it's magic because apprentice wizards know such things. The key is in location 7, the door to this location is simply kicked in as it is rotten. The apprentice also finds an ancient jug which will hold the water found in location 5, dripping from the ceiling.

All the apprentice needs to do is to put the seed in the hole found in the floor of location 1, water it and then climb the plant to freedom.

As you can see, the game is quite simple, really just an opening phase to a much larger game. Perhaps once free, the apprentice goes back to his village, collects useful items like more magic words, a rope and suchlike. A rope will be useful because I would have the plant wither and die after a number of moves. The rope will need to be tied to the tree near the hole and dangling into the pit or else there will be no escape! Nasty, am I not? Anyway, properly equipped the apprentice can delve deeper into this lost but strangely familiar world...

So much for the plot. That was the artistic and exciting bit. Now comes the down to earth but challenging task of transferring the idea to the computer. First, a list of locations, those in brackets are

possible exits, but are still set to zero at the start of play. These exits will be saved when we eventually get to the save and load game routines. The question mark means that the exit may or may not exist, depending whether the player has wedged the secret door open or not.

Location	Description	Exits
1	A stone pit	(U),(E)
2	A dark passage	S, ?E
3	Square chamber	N,S,E,(W)
4	Storeroom	E
5	Dead end passage	N
6	Twisty passage	W,(N)
7	Small chamber	S
8	Outside	D

Next a list of objects and starting locations.

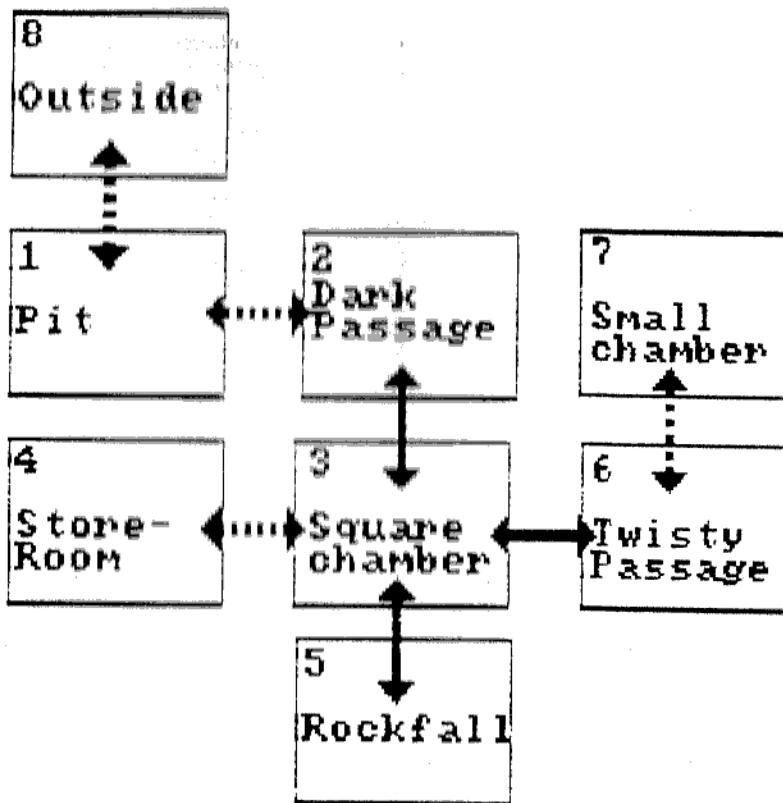
Object	Description	Location	Use
1	Magic staff	999	Find secret door, light
2	Rock	1	Wedge secret door open
3	Seed packet	4	Contains magic seed
4	Magic seed	0	Escape location 1
5	Large key	7	Unlock door in 3
6	Empty jug	7	Container for water
7	Broken chair	7	Scenery
8	Empty seed packet	0	None
9	Full jug	0	Water seed

These are the "takeable objects". As you can see, the first seven are 'real' objects, with the last two being swapped in and out of the game as required. You might call the last two virtual objects.

Now it's your turn to do some work. Using the location editor you have hopefully written, type in suitable atmospheric descriptions for the location descriptions along with the exits. Remember, for location 1 to mention the hole in the floor and the fact that the opening in the roof is strangely regular (the shape is up to you!) And don't forget the water dripping from the roof of location 5!

Next you can write the object descriptions (OD\$(9), the examine messages EX\$(9)-remember the magic word "vulco" written on the magic staff and the fact it has a crystal on the top! The object condition array can be filled in, remembering objects 4,8 and 9 do not exist yet, i.e, OC(x,0) and OC(x,1) are both zero. The staff is of course carried by the player.

The next article will contain the verb list along with the first part of the program listing.

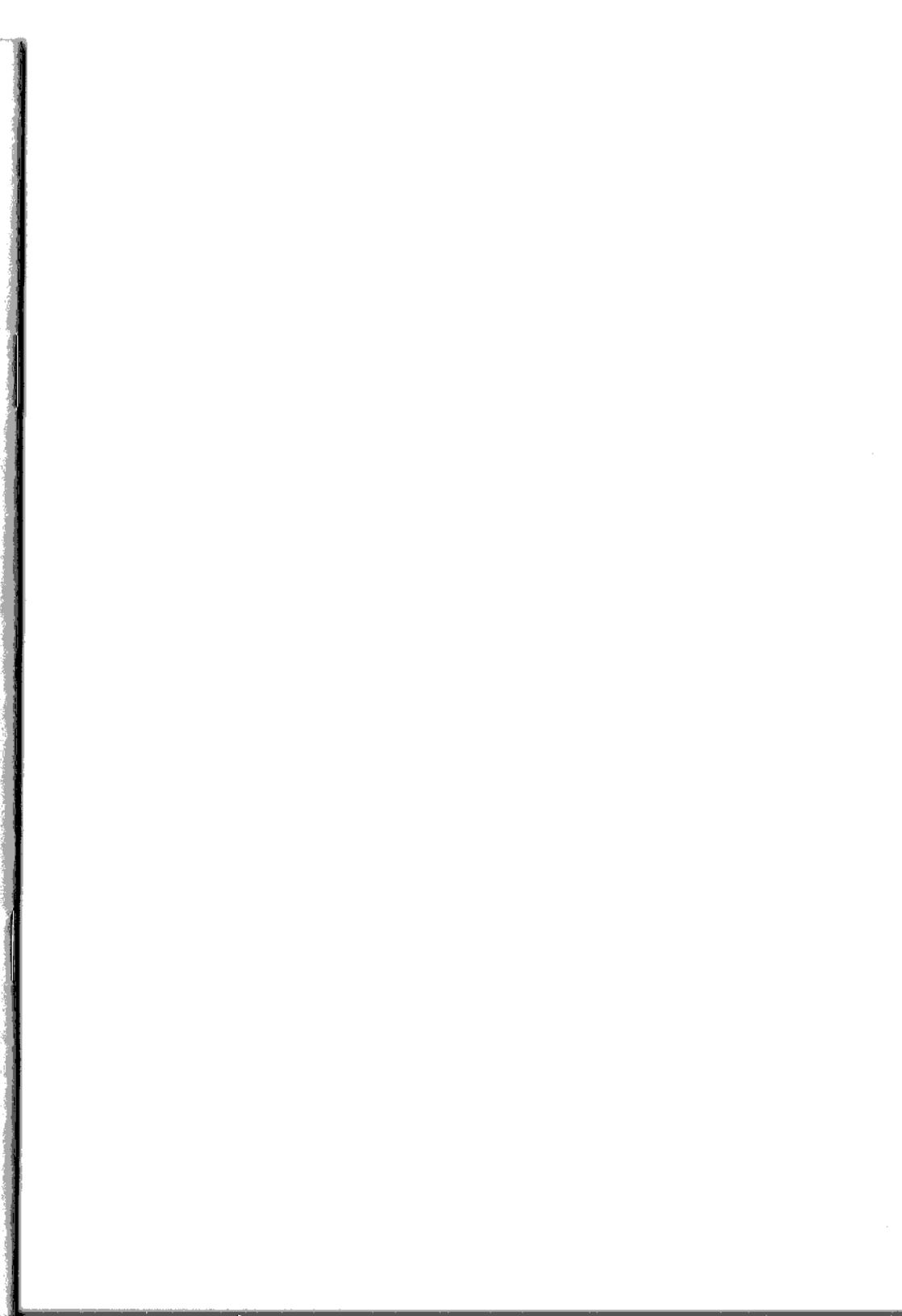


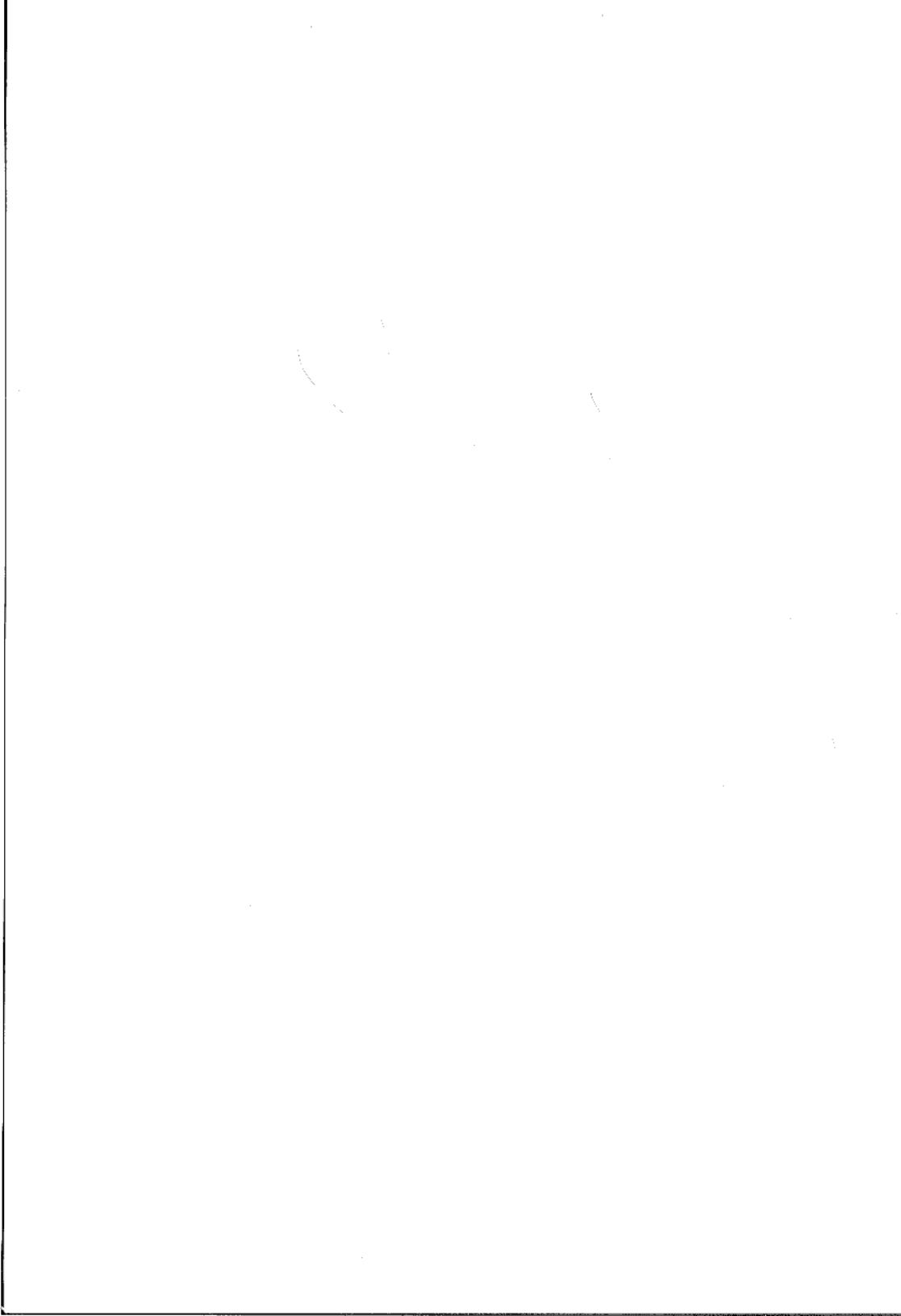
PAW TECHNICAL GUIDE

by Steve Clay

This is an index for the technical manual for PAW. Gilsoft omitted to include one.

CONDACT	PAGE	CONDACT	PAGE	CONDACT	PAGE
<b>ABILITY</b>	22	<b>EQ</b>	12	<b>PICTURE</b>	26
<b>ABSENT</b>	12	<b>EXTERN</b>	32	<b>PLACE</b>	16
<b>ADD</b>	21	<b>GET</b>	14	<b>PLUS</b>	21
<b>ADJECT1</b>	13	<b>GOTO</b>	22	<b>PREP</b>	13
<b>ADJECT2</b>	13	<b>GRAPHIC</b>	28	<b>PRESENT</b>	12
<b>ADVERB</b>	13	<b>GT</b>	13	<b>PRINT</b>	25
<b>ANYKEY</b>	30	<b>INK</b>	26	<b>PRINTAT</b>	27
<b>AT</b>	11	<b>INPUT</b>	24	<b>PROCESS</b>	31
<b>ATGT</b>	11	<b>INVEN</b>	27	<b>PROMPT</b>	24
<b>ATLT</b>	11	<b>ISAT</b>	12	<b>PROTECT</b>	25
<b>AUTOD</b>	18	<b>ISNOTAT</b>	12	<b>PUTIN</b>	16
<b>AUTOG</b>	18	<b>LET</b>	21	<b>PUTO</b>	16
<b>AUTOP</b>	19	<b>LINE</b>	23	<b>QUIT</b>	14
<b>AUTOR</b>	19	<b>LISTAT</b>	27	<b>RAMLOAD</b>	29
<b>AUTOT</b>	19	<b>LISTOBJ</b>	27	<b>RAMLOAD</b>	29
<b>AUTOW</b>	18	<b>LOAD</b>	29	<b>RANDOM</b>	21
<b>BACKAT</b>	27	<b>LT</b>	13	<b>REMOVE</b>	15
<b>BEEP</b>	31	<b>MES</b>	26	<b>RESET</b>	32
<b>BORDER</b>	26	<b>MESSAGE</b>	26	<b>SAME</b>	13
<b>CARRIED</b>	12	<b>MINUS</b>	21	<b>SAVE</b>	29
<b>CHARSET</b>	26	<b>MODE</b>	22	<b>SAVEAT</b>	27
<b>CHANCE</b>	14	<b>MOVE</b>	21	<b>SCORE</b>	25
<b>CLEAR</b>	20	<b>NEWLINE</b>	26	<b>SET</b>	20
<b>CLS</b>	26	<b>NEWTEXT</b>	31	<b>SUB</b>	21
<b>COPYFF</b>	21	<b>NOTAT</b>	11	<b>SWAP</b>	16
<b>COPYFO</b>	20	<b>NOTCARR</b>	12	<b>SYSMESS</b>	26
<b>COPYOF</b>	20	<b>NOTDONE</b>	28	<b>TAKEOUT</b>	17
<b>COPYOO</b>	19	<b>NOTEQ</b>	13	<b>TIME</b>	24
<b>CREATE</b>	16	<b>NOTSAME</b>	13	<b>TIMEOUT</b>	14
<b>DESC</b>	28	<b>NOTWORN</b>	12	<b>URNS</b>	25
<b>DESTROY</b>	16	<b>NOTZERO</b>	12	<b>WEAP</b>	15
<b>DOALL</b>	31	<b>NOUN2</b>	13	<b>WEIGH</b>	20
<b>DONE</b>	28	<b>OK</b>	28	<b>WEIGHT</b>	22
<b>DROP</b>	15	<b>PAPER</b>	26	<b>WHATO</b>	20
<b>DROPALL</b>	17	<b>PARSE</b>	30	<b>WORN</b>	12
<b>END</b>	28	<b>PAUSE</b>	30	<b>ZERO</b>	12





DEAR ALL,

PLEASE FIND ENCLOSED THE LATEST ISSUE OF "ADVENTURE CODER". IT'S LATE... SO WHAT HAPPENED?! I SENT IT TO MY USUAL PUBLISHER BUT, AFTER A WHILE, IT WAS RETURNED! APPARENTLY THEY CAN NO LONGER PUBLISH MAGAZINES. SO I WAS ON THE LOOK-OUT FOR A NEW PUBLISHER, AND MY OLD ONE SUGGESTED I WROTE TO "ADVENTURE PROBE" TO SEE IF THEIR NEW PUBLISHER WOULD BE WILLING TO TAKE ON "CODER"... AFTER SOME NEGOTIATING, HE AGREED! BUT THEN HE WAS ABOUT TO MOVE HOUSE, SO I WAITED SOME MORE! WHAT'S MORE, I'M ABOUT TO MOVE HOUSE, BUT IT KEEPS GETTING DELAYED. I'LL LET EVERYONE KNOW MY NEW ADDRESS WHEN I DO! UNTIL THEN, ENJOY THIS MUCH-DELAYED ISSUE, AND I APOLOGISE FOR IT'S LATENESS!

Christopher.